

CSE 160 23wi Midterm Exam Cheat Sheet

```
# if/elif/else syntax
if condition1:
    # statements
elif condition2:
    # other statements
else:
    # more statements
```

```
# for loop syntax
for i in sequence:
    # statements

# function definition syntax
def function_name(param1, param2, ...):
    # statements
```

Function	Description
<code>range([start,] stop [, step])</code>	Returns a sequence of numbers from start (inclusive) to stop (exclusive) incremented by step
<code>len(Lst)</code>	Returns the number of elements in Lst

Lists

Function	Description
<code>lst = []</code>	Creates an empty list
<code>lst[idx]</code>	Returns the element in Lst at index idx
<code>lst[start : end]</code>	Returns a sublist of Lst from index start to index end (exclusive)
<code>lst[start : end : step]</code>	Returns a sublist of Lst from index start to index end (exclusive), incrementing by step
<code>lst.append(eLmt)</code>	Adds the element eLmt to the end of Lst . Returns None .
<code>lst.index(eLmt)</code>	Returns index of the first occurrence of eLmt in Lst , Error if eLmt is not in lst
<code>lst.count(eLmt)</code>	Returns the number of times eLmt occurs in Lst
<code>lst.remove(eLmt)</code>	Removes first occurrence of eLmt from Lst , Error if eLmt is not in Lst . Returns None .
<code>lst.pop(idx)</code> <code>lst.pop()</code>	Removes and returns the element at index idx in Lst . With no parameter, removes the last element in Lst
<code>lst.insert(idx, eLmt)</code>	Inserts an element eLmt in list at index idx . Returns None .

File I/O

Function	Description
<code>my_file = open(<i>filepath</i>)</code>	Opens the file with given <i>filepath</i> for reading, returns a file object
<code>my_file.close()</code>	Closes file <code>my_file</code>

```
# Process one line at a time:  
for line_of_text in my_file:  
    # process line_of_text
```

```
# Process entire file at once  
all_data_as_a_big_string = my_file.read()
```

Dictionaries

Function	Description
<code>my_dict = {}</code>	Creates a new, empty dictionary
<code>my_dict[key]</code>	Returns the value associated with the given key in <i>my_dict</i>
<code>my_dict.keys()</code>	Returns list of keys in <i>my_dict</i>
<code>my_dict.values()</code>	Returns list of values in <i>my_dict</i>

Sorting

Function	Description
<code>sorted(<i>collection</i> [,key=<i>sort_key</i>, reverse=<i>bool_val</i>])</code>	Returns a sorted copy of <i>collection</i> , based on optional sort key (key) and optional order preference (reverse)
<code><i>lst</i>.sort([key=<i>sort_key</i>, reverse=<i>bool_val</i>])</code>	Sorts the given list <i>lst</i> , based on optional sort key (key) and optional order preference (reverse), and returns None

Common Error Names

- IndexError – Index out of range
- KeyError – Key not found in dictionary
- IndentationError – Invalid indentation
- TypeError – Operation applied to invalid combination of types
- ValueError – Function gets properly typed argument, but invalid value
- SyntaxError – Invalid Python syntax
- NameError – Variable name not found
- FloatingPointError – Floating point operation fails
- RuntimeError – Otherwise Unknown Error